

CLAIMS

What is claimed is:

1 1. A method, comprising:
2 executing a pre-boot application within an emulated pre-boot environment
3 to test functionality of the pre-boot application, the emulated pre-boot
4 environment executing within a user mode of a processor of a processing system
5 during an operating system ("OS") runtime of the processing system; and
6 interacting with a hardware device of the processing system in response
7 to the executing of the pre-boot application via a kernel proxy agent executing in
8 a kernel mode of the processor.

1 2. The method of claim 1, further comprising reporting whether an error
2 occurred during the interacting with the hardware device.

1 3. The method of claim 1, further comprising executing an interface
2 translator in the user mode of the processor during the OS runtime to pass a
3 request for hardware interaction from the pre-boot application to a corresponding
4 OS user mode application programming interface ("API") if the corresponding OS
5 user mode API is provided by the OS and to pass the request for hardware
6 interaction to the kernel proxy agent if the OS does not provide the
7 corresponding OS user mode API.

1 4. The method of claim 3 wherein interacting with the hardware device via
2 the kernel proxy agent comprises publishing an interface to the user mode of the
3 processor by the kernel proxy agent, the interface to enable the interface
4 translator to pass the request for hardware interaction to the kernel proxy agent.

1 5. The method of claim 1, further comprising executing the kernel proxy
2 agent to reserve a hardware resource of the processing system for use by the
3 hardware device.

1 6. The method of claim 5 wherein the hardware device comprises a
2 peripheral component interconnect ("PCI") add-in card and wherein the hardware
3 resource includes a PCI slot for communicatively coupling the PCI add-in card
4 into the processing system.

1 7. The method of claim 1 further comprising copying the pre-boot
2 application into an option read only memory ("ROM") of the hardware device after
3 the functionality of the pre-boot application is determined to be correct.

1 8. The method of claim 7 wherein the pre-boot application comprises a
2 hardware driver of the hardware device.

1 9. A machine-accessible medium that provides instructions that, if
2 executed by a machine, will cause the machine to perform operations
3 comprising:
4 executing an emulated pre-boot environment to test functionality of a pre-
5 boot application, the emulated pre-boot environment executing within a user
6 mode of a processor of a processing system during an operating system ("OS")
7 runtime of the processing system; and
8 interacting with a hardware device of the processing system in response
9 to a request for hardware interaction from the pre-boot application, the interacting
10 via a kernel proxy agent executing in a kernel mode of the processor.

1 10. The machine-accessible medium of claim 9, further providing
2 instructions that, if executed by the machine, will cause the machine to perform
3 further operations, comprising:
4 reporting whether an error occurred during the interacting with the
5 hardware device.

1 11. The machine-accessible medium of claim 9, further providing
2 instructions that, if executed by the machine, will cause the machine to perform
3 further operations, comprising:
4 executing an interface translator in the user mode of the processor during
5 the OS runtime to pass the request for hardware interaction from the pre-boot
6 application to a corresponding OS user mode application programming interface

7 (“API”) if the corresponding OS user mode API is provided by the OS and to pass
8 the request for hardware interaction to the kernel proxy agent if the OS does not
9 provide the corresponding OS user mode API.

1 12. The machine-accessible medium of claim 11, further providing
2 instructions that, if executed by the machine, will cause the machine to perform
3 the operations wherein interacting with the hardware device via the kernel proxy
4 agent comprises publishing an interface to the user mode of the processor by the
5 kernel proxy agent, the interface to enable the interface translator to pass the
6 request for hardware interaction to the kernel proxy agent.

1 13. The machine-accessible medium of claim 9, further providing
2 instructions that, if executed by the machine, will cause the machine to perform
3 further operations, comprising:
4 executing the kernel proxy agent to reserve a hardware resource of the
5 processing system for use by the hardware device.

1 14. The machine-accessible medium of claim 13 wherein the hardware
2 device comprises a peripheral component interconnect (“PCI”) add-in card and
3 wherein the hardware resource includes a PCI slot for communicatively coupling
4 the PCI add-in card into the processing system.

1 15. The machine-accessible medium of claim 9, further providing
2 instructions that, if executed by the machine, will cause the machine to perform
3 further operations, comprising:
4 installing the kernel proxy agent into the kernel mode during an
5 initialization phase of the OS.

1 16. A processing system, comprising:
2 a processor to execute an operating system ("OS") and to execute a pre-
3 boot application, the processor having a user mode and a kernel mode;
4 a hardware device communicatively coupled to the processor; and
5 a data storage unit communicatively coupled to the processor and having
6 stored thereon a pre-boot environment module and a kernel proxy agent, the pre-
7 boot environment module to be executed by the processor to generate an
8 emulated pre-boot environment within the user mode for executing the pre-boot
9 application, the kernel proxy agent to be executed by the processor to enable
10 interaction between the pre-boot application and the hardware device when the
11 OS does not include an OS user mode application programming interface ("API")
12 for interacting with the hardware device.

1 17. The processing system of claim 16 wherein the kernel proxy agent
2 includes instructions to reserve a hardware resource of the processing system for
3 use by the hardware device.

1 18. The processing system of claim 17 wherein the hardware device
2 comprises a peripheral component interconnect ("PCI") add-in card and wherein
3 the hardware resource includes a PCI slot for communicatively coupling the PCI
4 add-in card to the processor.

1 19. The processing system of claim 16, further comprising the data
2 storage unit having stored thereon an interface translator to be executed by the
3 processor, the interface translator executable in the user mode to pass a request
4 for hardware interaction from the pre-boot application to a corresponding OS
5 user mode API if the corresponding OS user mode API is provided by the OS
6 and to pass the request to the kernel proxy agent if the OS does not provide the
7 corresponding OS user mode API.

1 20. The processing system of claim 19 wherein the kernel proxy agent is
2 further to be executed by the processor to generate an error report if the quest
3 for hardware interaction passed to the kernel proxy agent results in an error.